In the Claims:

Please amend the claims to read as indicated below.

1. (currently amended) A method of compiling a software program for a programmable processor having a functional unit associated with at least two issue slots, the method comprising:

receiving a set of processor-executable operations comprising a first processor-executable operation of a type typically associated with at least two issue slots and a second processor-executable operation; and

replacing the first processor-executable operation ~~of the type associated with at least two issue slots of the functional unit~~ with a third equivalent processor-executable operation associated with fewer than all of the issue slots, thereby allowing one or more of the rest of the issue slots to be used by the second processor-executable instruction.

2. (currently amended) The method of claim 1, wherein replacing the first processor-executable operation with a third equivalent processor-executable operation further ~~comprising~~ comprises analyzing the first processor-executable operation ~~of the type typically associated with at least two issue slots~~ and external information to determine whether the first processor-executable operation ~~of the type typically associated with at least two issue slots~~ can be replaced by the third equivalent ~~a~~ processor-executable operation ~~associated with fewer than all of the issue slots associated with the functional unit~~.

3. (currently amended) The method of claim 1, wherein ~~replacing the processor-executable instruction comprises replacing the processor-executable operation of the type typically associated with at least two issue slots with an~~ the third equivalent processor-executable operation ~~is~~ associated with only one issue slot.

4. (currently amended) The method of claim 1, wherein ~~replacing the processor-executable instruction comprises replacing the processor-executable operation of the type typically associated with at least two issue slots with an~~ the third equivalent processor-executable operation ~~is~~ associated with a plurality of issue slots.

2

5. (currently amended) The method of claim 1, <u>wherein replacing the first</u>
<u>processor-executable operation with a third equivalent processor-executable operation</u>
further ~~comprising~~ <u>comprises</u>:

determining a number of input registers and a number of output registers that are
used by the <u>first</u> processor-executable operation ~~of the type typically associated with at~~
~~least two issue slots~~; and

when the <u>first</u> processor-executable operation ~~of the type typically associated with~~
~~at least two issue slots~~ uses at most two input registers and one output register, replacing
the <u>first</u> processor-executable operation ~~of the type typically associated with at least two~~
~~issue slots~~ with ~~an~~ <u>the third</u> equivalent processor-executable operation associated with
only one issue slot.

6-7. (cancelled)

8. (currently amended) The method of claim 1, wherein the <u>first</u> processor-
executable operation ~~of the type typically associated with at least two issue slots~~ is a
shuffle operation.

9. (currently amended) The method of claim 1, wherein the <u>first</u> processor-
executable operation ~~of the type typically associated with at least two issue slots~~ is a
floating point operation.

10. (original) A method of compiling a software program for a programmable
processor having a functional unit associated with a plurality of issue slots, the method
comprising:

receiving a processor-executable superoperation of a type typically associated
with at least two issue slots;

determining a number of input registers and a number of output registers that are
used by the superoperation; and

when the superoperation uses at most two input registers and one output register,
replacing the superoperation with an equivalent processor-executable operation
associated with only one issue slot.

3

1   11. (original) The method of claim 10, further comprising:

2   identifying any source operations that produce a result affecting a result of the

3   superoperation;

4   placing commands for the source operations in instruction words;

5   selecting an earliest instruction word from a set of instruction words after the

6   instruction words in which commands for the source operations have already been

7   placed; and

8   determining whether an instruction word can be constructed that contains any

9   commands already included in the earliest instruction word in addition to a command for

10 the superoperation.

1   12. (original) The method of claim 11, further comprising:

2   when an instruction word that contains any commands already included in the

3   earliest instruction word in addition to the command for the superoperation cannot be

4   constructed, selecting a subsequent instruction word; and

5   determining whether an instruction word that contains any commands already

6   included in the earliest instruction word in addition to the command for the

7   superoperation can be constructed using the subsequent instruction word.

1   13. (original) The method of claim 10, wherein the superoperation is a shuffle

2   operation.

1   14. (original) The method of claim 10, wherein the superoperation is a floating

2   point operation.

1   15. (currently amended) A method of executing an a first instruction that is

2   typically associated with at least two issue slot by a processor having a functional unit

3   associated with a plurality of issue slots, the method comprising:

4   determining whether the first instruction can be executed using fewer than the at

5   least two all of the issue slots associated with the functional unit; and

6   when the first instruction can be executed using fewer than the at least two all of

7   the issue slots associated with the functional unit, mapping replacing the first instruction

8   with a second equivalent instruction associated with to fewer than the at least two all of

9   ~~the~~ issue slots thereby allowing one or more of the rest of the issue slots to be used by a

10   third instruction.

1   16. (cancelled)

1   17. (currently amended) The method of claim 15, wherein determining whether

2   the first instruction can be executed using fewer than the at least two issue slots further

3   comprises ~~comprising~~:

4   determining a number of input registers and a number of output registers that are

5   used by the first instruction; and

6   when the first instruction uses at most two input registers and one output register,

7   replacing ~~mapping~~ the first instruction with the second instruction, the second instruction

8   being associated with ~~to~~ a single issue slot.

1   18-19. (cancelled)

1   20. (currently amended) The method of claim 15, wherein the second instruction

2   is associated with ~~further comprising mapping the instruction to~~ two or more ~~a plurality~~

3   ~~of~~ issue slots.

1   21. (currently amended) The method of claim 15, wherein the first instruction is a

2   shuffle operation.

1   22. (currently amended) The method of claim 15, wherein the first instruction is a

2   floating point operation.

1   23. (currently amended) A processor-readable medium containing processor-

2   executable instructions for:

3   receiving a set of operations comprising a first operation of a type typically

4   associated with at least two issue slots of a functional unit of a programmable processor

5   and a second operation; and

6   replacing the first operation of the type typically associated with at least two issue

7   slots by a third equivalent operation associated with fewer than all of the issue slots

8   associated with the functional unit, thereby allowing one or more of the rest of the issue

9   slots to be used by the second operation.

5

1   24. (currently amended) The processor-readable medium of claim 23, further

2   containing processor-executable instructions for analyzing the first operation ~~of the type~~

3   ~~typically associated with at least two issue slots~~ and external information to determine

4   whether the first operation ~~of the type typically associated with at least two issue slots~~

5   can be replaced by ~~a~~ the third equivalent operation ~~associated with fewer than all of the~~

6   ~~issue slots associated with the functional unit.~~

1   25. (currently amended) The processor-readable medium of claim 23, ~~further~~

2   ~~containing processor-executable instructions for replacing the operation of the type~~

3   ~~typically associated with at least two issue slots with~~ wherein the third ~~an~~ equivalent

4   operation is associated with only one issue slot.

1   26. (currently amended) The processor-readable medium of claim 23, further

2   containing processor-executable instructions for:

3   determining a number of input registers and a number of output registers that are

4   used by the first operation ~~of the type typically associated with at least two issue slots~~;

5   and

6   when the first operation ~~of the type typically associated with at least two issue~~

7   ~~slots~~ uses at most two input registers and one output register, replacing the first operation

8   ~~of the type typically associated with at least two issue slots~~ with ~~an~~ the third equivalent

9   operation associated with only one issue slot.

1   27-28. (cancelled)

1   29. (currently amended) The processor-readable medium of claim 23, wherein

2   the first operation ~~of the type typically associated with at least two issue slots~~ is a shuffle

3   operation.

1   30. (currently amended) The processor-readable medium of claim 23, wherein

2   the first operation ~~of the type typically associated with at least two issue slots~~ is a floating

3   point operation.

1   31. (original) A processor-readable medium containing processor-executable

2   instructions for:

3   receiving a superoperation of a type typically associated with at least two issue

4   slots of a functional unit of a programmable processor;

5   determining a number of input registers and a number of output registers that are

6   used by the superoperation; and

7   when the superoperation uses at most two input registers and one output register,

8   replacing the superoperation with an equivalent operation associated with only one issue

9   slot.

1   32. (original) The processor-readable medium of claim 31, further containing

2   processor-executable instructions for:

3   identifying any source operations that produce a result affecting a result of the

4   superoperation;

5   placing commands for the source operations in instruction words;

6   selecting an earliest instruction word from a set of instruction words after the

7   instruction words in which commands for the source operations have already been

8   placed; and

9   determining whether an instruction word can be constructed that contains any

10 commands already included in the earliest instruction word in addition to a command for

11 the superoperation.

1   33. (original) The processor-readable medium of claim 32, further containing

2   processor-executable instructions for:

3   when an instruction word that contains any commands already included in the

4   earliest instruction word in addition to the command for the superoperation cannot be

5   constructed, selecting a subsequent instruction word; and

6   determining whether an instruction word that contains any commands already

7   included in the earliest instruction word in addition to the command for the

8   superoperation can be constructed using the subsequent instruction word.

1   34. (original) The processor-readable medium of claim 31, wherein the

2   superoperation is a shuffle operation.

1   35. (original) The processor-readable medium of claim 31, wherein the

2   superoperation is a floating point operation.

7

1 36. (currently amended) A processor-readable medium containing processor-

2 executable instructions for:

3 determining whether a first ~~the~~ instruction that is typically associated with at least

4 two issue slots in a functional unit of a processor can be executed using fewer than the at

5 least two ~~all~~ issue slots ~~associated with a functional unit of a processor~~; and

6 when the first instruction can be executed using fewer than the at least two ~~all~~

7 issue slots ~~associated with the functional unit~~, replacing ~~mapping~~ the first instruction with

8 a second equivalent instruction that is associated with ~~to~~ fewer than the at least two ~~all~~

9 issue slots thereby allowing one or more of the rest of the issue slots to be used by a third

10 instruction.

1 37. (currently amended) The processor-readable medium of claim 36, further

2 containing processor-executable instructions for analyzing the first instruction and

3 external information to determine whether the first instruction can be executed using

4 fewer than the at least two ~~all~~ issue slots ~~associated with the functional unit~~.

1 38. (currently amended) The processor-readable medium of claim 36, further

2 containing processor-executable instructions for:

3 determining a number of input registers and a number of output registers that are

4 used by the first instruction; and

5 when the first instruction uses at most two input registers and one output register,

6 replacing ~~mapping~~ the first instruction with the second instruction, the second instruction

7 being associated with ~~to~~ a single issue slot.

1 39-40. (cancelled)

1 41. (currently amended) The processor-readable medium of claim 36, wherein

2 the second instruction is associated with ~~further containing processor-executable~~

3 ~~instructions for mapping the instruction to~~ two or more ~~a plurality of~~ issue slots.

1 42. (currently amended) The processor-readable medium of claim 36, wherein

2 the first instruction is a shuffle operation.

43. (currently amended) The processor-readable medium of claim 36, wherein the first instruction is a floating point operation.

44. (new) A method for compiling a software program for a programmable processor having a functional unit associated with at least two issue slots, the method comprises:

identifying source operations from a set of processor-executable operations that produce operands of a first processor-executable operation;

placing commands for the source operations in one or more instruction words;

selecting an earliest instruction word from a set of instruction words after the one or more instruction words in which commands for the source operations have already been placed;

determining whether the first processor-executable operation can be replaced by a second equivalent processor-executable operation;

replacing the first processor-executable operation with the second equivalent processor-executable operation if it is determined that the first processor-executable operation can be so replaced;

determining whether an instruction word can be constructed that contains the commands already included in the earliest instruction word and commands for the first processor-executable operation or the second processor-executable operation if it has replaced the first processor-executable operation; and

if such an instruction word can be constructed, updating the earliest instruction word to include the commands for the first processor-executable operation or the second processor-executable operation if it has replaced the second processor-executable operation.

45. (new) The method of claim 44, further comprising:

if such an instruction word cannot be constructed, selecting an subsequent instruction word after the earliest instruction word; and

determining whether an instruction word can be constructed that contains the commands already included in the subsequent instruction word and commands for the

9

6   first processor-executable operation or the second processor-executable operation if it has

7   replaced the first processor-executable operation.

1   46. (new) A processor-readable medium having processor-executable instructions

2   for:

3   identifying source operations from a set of processor-executable operations that

4   produce operands of a first processor-executable operation;

5   placing commands for the source operations in one or more instruction words;

6   selecting an earliest instruction word from a set of instruction words after the one

7   or more instruction words in which commands for the source operations have already

8   been placed;

9   determining whether the first processor-executable operation can be replaced by a

10   second equivalent processor-executable operation;

11   replacing the first processor-executable operation with the second equivalent

12   processor-executable operation if it is determined that the first processor-executable

13   operation can be so replaced;

14   determining whether an instruction word can be constructed that contains the

15   commands already included in the earliest instruction word and commands for the first

16   processor-executable operation or the second processor-executable operation if it has

17   replaced the first processor-executable operation; and

18   if such an instruction word can be constructed, updating the earliest instruction

19   word to include the commands for the first processor-executable operation or the second

20   processor-executable operation if it has replaced the second processor-executable

21   operation.

1   47. (new) The processor-readable medium of claim 46, having further processor-

2   executable instructions for:

3   if such an instruction word cannot be constructed, selecting an subsequent instruction

4   word after the earliest instruction word,; and

5   determining whether an instruction word can be constructed that contains the

6   commands already included in the subsequent instruction word and commands for the first

7   processor-executable operation or the second processor-executable operation if it has

8   replaced the first processor-executable operation.